Capítulo 1

SVG

Este capítulo trata sobre el desarrollo de imagenes SVG. En su esencia esta compuesto por tres partes: Fundamentos, Plataforma y Lenguaje de programación SVG.

1.1. Fundamentos

1.1.1. Terminología

SVG o Vector de Gráficos escalable, es un lenguaje de marcado estándar W3C [1, p. 577], para describir gráficos bidimensionales en XML y se convirtió en una recomendación del W3C en septiembre de 2001.

1.1.2. Versión

Las distintas versiones SVG de la *W3C Recommendation*, informes de especificación técnica elaboradas por los miembros de *SVG Working Group*.

SVG 1.0 (Jon Ferraiolo) [2] 2001-09-04.
SVG 1.0, última edición:
`https://www.w3.org/TR/SVG10/`

SVG 1.1 (First Edition) 2003-01-14.

SVG Tiny 1.2 2008-12-22.
El SVG Working Group considera la versión Tiny 1.2 como una rama desaprobada del estándar SVG [3].

SVG 1.1 (Second Edition) [4] 2011-08-16.
SVG 1.1, última edición:
`https://www.w3.org/TR/SVG11/`.

SVG 2 (Draft) 2014–2015.
2014-02-11, 2015-04-09, 2015-09-15;

SVG 2 (Candidate) 2016–2018.
2016-09-15, 2018-08-07, 2018-10-04 [5].
SVG, última edición:
`https://www.w3.org/TR/SVG/`.

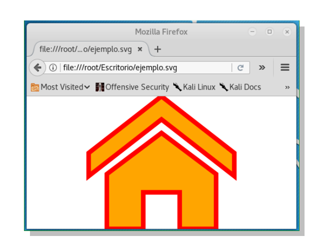
La lista fue mapeada tomando como base a [6, p. 2], y consolidada según lo declarado en los campos *this* y *previous version* de la especificación (§1.7.2) [5].

1.1.3. Cualidades

Las imágenes SVG se emplean en la creación de logos e iconos con efectos especiales. Las imágenes SVG son dinámicas (a diferencia de PNG o JPG) y pueden moverse, asignarles efectos CSS (§3), como un brillo cada cierto tiempo.

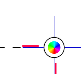
Figura 1.1. SVG Standard





Ventajas: HTML5 Standard (W3C) Figura 1.1; Tamaño pequeño; Alta calidad y escalable; Soporta JS event handlers; A diferencia de los PNG o JPG, los dibujos vectoriales no pierden calidad a medida que se les hace zoom. Por esta cualidad se les llama escalables.
Desventajas: No esta preparado para game applications [7]; Renderización lenta cuando el dibujo vectorial es muy complejo (uso de DOM profundo), y algunas veces la visualización tarda mucho, a diferencia de una imagen png o jpg donde el procesador no tiene que realizar cálculo alguno.

PRINT PREVIEW (BETA TEST) ~ PRESENTACIÓN PRELIMINAR

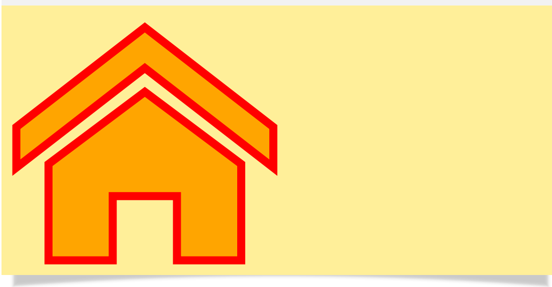


1.1.4. Código fuente

File extension *.svg

Un archivo con esta extensión contiene código fuente en lenguaje SVG.

```
<svg xmlns="http://www.w3.org/2000/svg"
     viewBox="0 0.967 32 29.033">
  <path stroke="red"
        fill="orange"
        d="M 32 18.451 16 6.031 0 18.451
           V 13.387 L 16 0.967 32 13.387 Z
           M 28 18 V 30 H 20 V 22 H 12
           V 30 H 4 V 18 L 16 9 Z" />
</svg>
```



Para divisar una imagen SVG puede usar un Navegador Firefox o Chrome, solamente arrastre el archivo hacia el navegador (Figura 1.1).

1.1.5. Edición de imagen SVG

Si quiere editar una imagen sin tocar código, puede optar por usar softwares que manipulen imágenes vectoriales como ser: Adobe Photoshop CC 2018 (Fig. 1.2), Inkscape 0.48 (Fig. 1.3).

Figura 1.2. Adobe Photoshop CC 2018

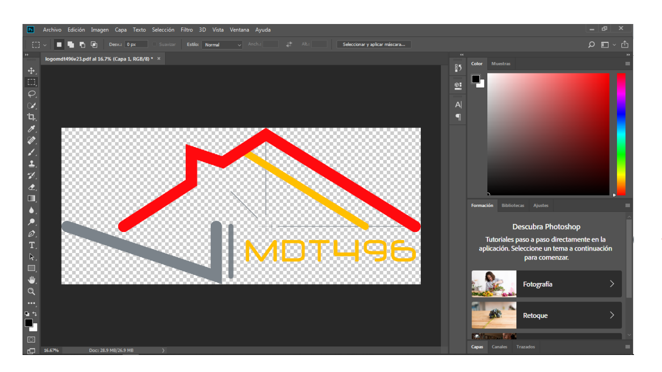
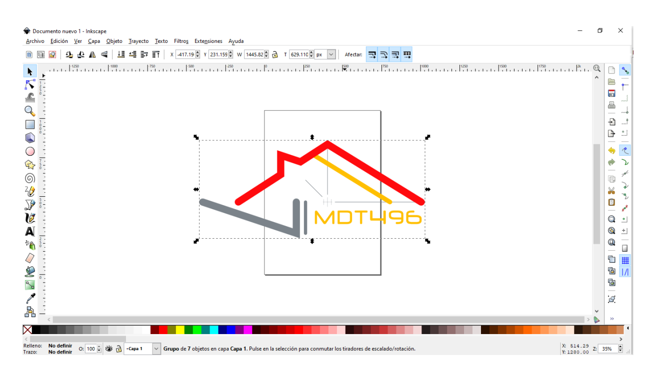


Figura 1.3. Inkscape 0.48.2 r9819



1.2. Estructura léxica

1.2.1. Elemento svg

<svg ⟨*opciones*⟩>
  ⟨*contenido del elemento*⟩
</svg>

Este elemento actúa como un contenedor y como un elemento estructural.

En el ⟨*contenido del entorno*⟩ se escriben las instruciones (una serie de elementos) que sirven para producir la figura formateada con las ⟨*opciones*⟩ especificadas.

```
<svg xmlns="http://www.w3.org/2000/svg"
    viewBox="-20 -10 90 40">
  <line x1="-20" y1="-10"
        x2="70" y2="30" stroke="red"/>
</svg>
```

del código anterior, se establece el atributo (de CSS) `stroke`. ya que por default no pinta la línea (transparente) y probablemente no logre visualizar la imagen.

Cada forma es un objeto, con elementos y atributos, y accesible en la DOM.

1.2.2. Comentario

<!--*x*--> (group operators)

Comentario de SVG, hereda el de XML.

```
<svg xmlns="http://www.w3.org/2000/svg">
  <!-- Lorem
       ipsum-->
```

Capítulo 2

HTML

2.1. Fundamentos

2.1.1. Terminología

HyperText Markup Language (HTML) is the publishing language used by the World Wide Web [9, §2.2, p. 21].

2.1.2. Versión

Las distintas versiones de la especificación HTML:

1995-11 HTML 2.0, RFC 1866 [10],
`https://doi.org/10.17487/RFC1866` (Obsoleted-By RFC2854)

1996-05 HTML Tables, RFC 1942 [11],
`https://doi.org/10.17487/RFC1942` (Obsoleted-By RFC2854)

1997-01 HTML (Internationalization of the), RFC 2070
`https://doi.org/10.17487/RFC2070` (Obsoleted-By RFC2854).

2000-06 The 'text/html' Media Type, RFC 2854 [12]
`https://doi.org/10.17487/RFC2854` (Obsoletes RFC2070, RFC1980, RFC1942, RFC1867, RFC1866).

1995-09 HTML 3.0,
`http://www.w3.org/MarkUp/html3/CoverPage`

1997-01-14 HTML 3.2
`https://www.w3.org/TR/REC-html32`

1997-03-24 HTML (and Style Sheets),
`http://www.w3.org/TR/WD-style-970324`

1998-04-24 HTML 4.0,
`http://www.w3.org/TR/1998/REC-html40-19980424`

1999-12-24 HTML 4.01, [9]
`https://www.w3.org/TR/1999/REC-html401-19991224/`

2014-10-28 HTML 5 (Living Standard),
https://html.spec.whatwg.org/multipage/

2016-11-01 HTML 5.1,
https://www.w3.org/TR/2016/REC-html51-20161101/

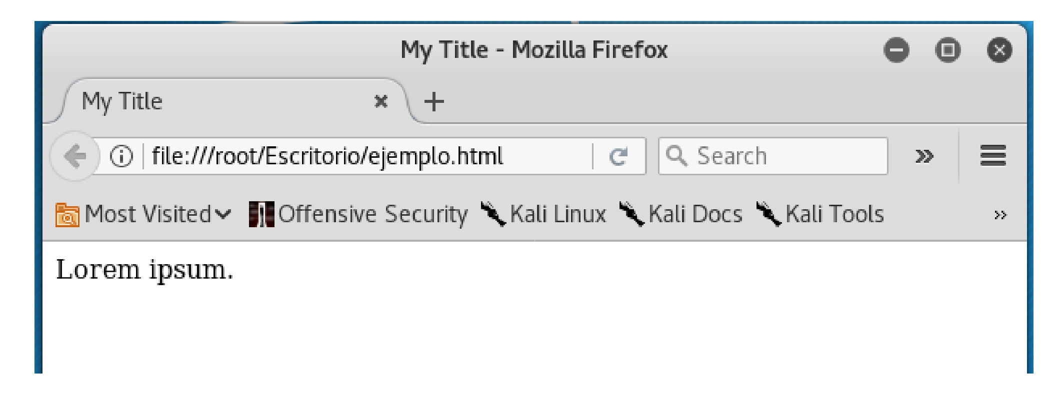
2.1.3. Código fuente

File extension *.html

Un archivo con esta extensión contiene código HTML, que puede ser interpretado por un navegador web (Firefox, Google Chrome, etc.). Por ejemplo (Figura 2.1):

```
<html>
    <head>
        <title>My Title</title>
    </head>
    <body>
        Lorem ipsum.
    </body>
</html>
```

Figura 2.1. HTML document



2.2. Estructura léxica

2.2.1. Esquema

La estructura del código fuente (estándar HTML5), en esencia, inicia con la declaración de la versión <!DOCTYPE> y luego se abre la página con el elemento <html>→P.22 (la raiz del documento).

```
<html ⟨opciones⟩>
  ⟨contenido del elemento⟩
</html>
```

Elemento [9, §7.3, p. 61] raíz de un documento HTML.

El ⟨*contenido del entorno*⟩ se compone por dos elementos: <head>→P.23 (*preámbulo*) y <body>→P.23 (*cuerpo*).

Las ⟨*opciones*⟩ son: lang y version.

Capítulo 3

CSS

3.1. Fundamentos

3.1.1. Terminología

Cascading Style Sheets (CSS) u Hoja de Estilos en Cascada, es un lenguaje de programación que sirve para escribir parte de la arquitectura de una aplicación web [1, Fig. 14.2, p. 573] donde, mediante atributos de estilo, permite definir el formato (la presentación visual [14, p. 7]) del documento HTML o XML (incluyendo SVG).

3.1.2. Versión

La especificación del lenguaje CSS (`https://www.w3.org/TR/CSS/#css`).

1998-03 (RFC2318) The text/css Media Type H. Lie, B. Bos, C. Lilley
`ftp://ds.internic.net/rfc/rfc2318.txt`

1996-12-17 CSS 1 (Cascading Style Sheets, level 1) [15]
`http://www.w3.org/TR/REC-CSS1-961217.html`

2008-04-11 CSS 2 (Cascading Style Sheets, level 2) [16]
`https://www.w3.org/TR/2008/REC-CSS2-20080411`

3.1.3. Elemento `style`

`<style>`
⟨*contenido del elemento*⟩
`</style>`

En el cuerpo de este elemento puede escribir código CSS. Opcionalmente puede declarar `type="text/css"`.

```
<html>
    <head>
        <title></title>
        <style type="text/css">
            p { color: red; }
        </style>
```

```
    </head>
    <body>
        <p>Lorem ipsum</p>
    </body>
</html>
```

Lorem ipsum

File extension `*.css`

El código CSS también puede estar en un archivo independiente (Secuencia 3.1), separado del documento HTML, y puede cargarse en el HTML mediante el atributo `href` de un elemento `<link>`.

Secuencia 3.1. Archivo `style.css`

```
p { color: red; }
```

```
<html>
    <head>
        <title></title>
        <link rel="stylesheet" href="styles.css">
    </head>
    <body>
        <p>Lorem ipsum</p>
    </body>
</html>
```

Lorem ipsum

3.2. Estructura léxica

3.2.1. Comentarios

`/*x*/` (group operators)

Comentario de bloque, donde todos los caracteres entre `/*` y `*/` son ignorados por el interprete.

```
/* Esto es un
       comentario en CSS */
```

3.2.2. Visibilidad (Styling)

`{x}` (group operators)

Las *opciones* de estilo CSS (para aplicar a un elemento) tienen alcance local al grupo.

```
body {
    padding: 0px;
    margin: 0px;
    font-family: Arial;
}
```

en el ejemplo, las opciones usadas son `padding`→P.49, `margin`→P.49 y `font-family`→P.46.